

LIST OF CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application.

Claims 1-48 (Cancelled)

49. (New) A method for inserting a data structure into a tree structure, comprising:
- examining a leaf node of the tree structure associated with a head portion of the data structure;
 - identifying a leaf node index associated with the head portion;
 - copying the leaf node index to a leaf node of the tree structure associated with a tail portion of the data structure;
 - determining a direction in which the data structure is traversing the tree structure;
 - updating indices within the tree structure according to an order based upon the direction in which the data structure is traversing the tree structure; and
 - deleting contents of the head portion of the data structure.
50. (New) The method of claim 49, further comprising:
- moving the tail portion of the data structure according to a pre-calculated increment.

51. (New) The method of claim 49, further comprising:
re-distributing empty leaf nodes within the tree structure among non-empty leaf nodes of the data structure.
52. (New) The method of claim 49, wherein if the direction in which the data structure is traversing the tree structure is towards a beginning of the tree, then the method includes,
updating the indices between the tail portion and a nearest non-empty leaf node from a lowest tree level to a highest tree level.
53. (New) The method of claim 52, further comprising:
updating a remainder of the tree structure from the highest tree level to the lowest tree level.
54. (New) The method of claim 49, wherein if the direction in which the data structure is traversing the tree structure is towards an end of the tree, then the method includes,
updating the indices from a lowest tree level to a highest tree level.
55. (New) A method for inserting data into a sorted tree structure, comprising method operations of:
providing a data structure marked by a head and a tail;

determining if the head of the data structure is associated with an end of the tree structure;

swapping the head and the tail if the head of the data structure is associated with the end of the tree structure;

determining an amount of empty space distributed among non-empty leaf nodes of the tree structure;

traversing the tree structure with the data structure in a traveling direction; and
maintaining the amount of empty space distributed among the non-empty leaf nodes while traversing the tree structure.

56. (New) The method of claim 55, wherein the method operation of determining an amount of empty space distributed among non-empty leaf nodes of the tree structure includes,

calculating an increment amount.

57. (New) The method of claim 56 wherein the increment amount includes a fractional portion.

58. (New) The method of claim 55, wherein the method operation of traversing the tree structure with the data structure in a traveling direction, includes:

moving the head of the data structure a single leaf node in the traveling direction.

59. (New) The method of claim 55, further comprising:
updating indices within the tree structure according to an order based upon the
traveling direction.

60. (New) The method of claim 59, wherein the order is from a lowest tree
level to a highest tree level when the traveling direction is towards an end of the tree
structure.

61. (New) The method of claim 59, wherein the order is from a highest tree
level to a lowest tree level when the traveling direction is towards a beginning of the tree
structure.